Transmission by Facsimile on May 14, 2007

PATENT
DKT. STL11422

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re application of: **Robert G. Bean, Clark E. Lubbers and Randy L. Roberson**
Assignee: **SEAGATE TECHNOLOGY LLC**
Application No.: **10/669,196** Group No.: **2113**
Filed: **September 23, 2003** Examiner: **Emerson Puente**
For: **DATA RELIABILITY BIT STORAGE QUALIFIER AND LOGICAL UNIT METADATA**

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P. O. Box 1450
Alexandria, Virginia 22313-1450

ATTENTION: Board of Patent Appeals and Interferences

APPELLANT'S REPLY BRIEF

Mitchell K. McCarthy, Reg. No. 38,794
McCarthy & Associates
500 West Main Street, Suite 609
Oklahoma City, Oklahoma 73102
Telephone: (405) 601-2798
Facsimile: (405) 601-2853

CERTIFICATION UNDER 37 C.F.R. §§ 1.8(a)

I hereby certify that, on the date shown below, this correspondence is being:

TRANSMISSION

☒ facsimile transmitted to the Patent and Trademark Office, (571) 273 - 8300.

Date: May 14, 2007

Jenna R. McCarthy
Signature

Jenna R. McCarthy
(type or print name of person certifying)

TABLE OF CONTENTS

RECEIVED
CENTRAL FAX CENTER
MAY 14 2007

A. INTRODUCTION

The present embodiments contemplate a computer system that stores user data with associated metadata indicating the status of the user data. Storing the two different types of data together enables the system to subsequently execute only one access command in order to both assess the reliability of the user data and retrieve the user data. This increases the system processing overhead efficiency by reducing the number of necessary seeks in comparison to other attempted solutions (like the cited reference[1] used to reject the claimed embodiments) where the user data and the metadata are stored separately and where mapping the user data to its associated metadata is required. Also, by using the metadata to directly indicate the user data status, the system is able to preempt an occurrence of cascading errors that result from processing unreliable user data, unlike other attempted solutions (like the cited reference[2] used to reject the claimed embodiments) where the metadata is employed for other purposes.

To decide this appeal the Board must determine whether the Examiner has substantiated determinative evidence that Lubbers '643 and/or Morgan '411 identically discloses all the elements of the claimed embodiments.[3] Ultimately, the prosecution process and this appeal must not sway from safeguarding the public notice function provided by the patent through the language of the claims and the specification.[4] Applicant's stated position is that these rights are properly safeguarded to all by the language of the claims on appeal, and that the Examiner's rejections are reversible error because they are based either on claim constructions that are not

[1] Lubbers '643
[2] Morgan '411
[3] *Verdegaal Bros. v. Union Oil Co. of California*, 2 USPQ 2d 1051, 1053 (Fed. Cir. 1987).
[4] *University of Rochester v. G.D. Searle & Co.*, 358 F.3d 916, 922 n.5 (Fed. Cir. 2004).

within the broadest reasonable interpretation consistent with the specification or a mischaracterization of what the cited reference discloses.[5]

B. ARGUMENTS

THE SECTION 102 REJECTION OF INDEPENDENT CLAIMS 1, 8, AND 15 OVER LUBBERS '643 IS CLEAR ERROR BECAUSE THE EVIDENCE IN THE RECORD IS DETERMINITIVE THAT THE REFERENCE DOES NOT DISCLOSE ALL THE RECITED FEATURES OF THESE CLAIMS

Claim 1

Claim 1 recites *storing first information with first data*. Applicant's position is that the broadest reasonable interpretation of this disputed claim phrase is that *with* means the *first information* and the *first data* are stored together.

During examination claims are given their "broadest reasonable interpretation consistent with the specification."[6] The "broadest reasonable interpretation" is the meaning that the skilled artisan would give to the claim term in view of the associated usage provided in the specification.[7] A construction that is inconsistent with the written description would not be arrived at by the skilled artisan, and is therefore not a "reasonable interpretation."[8]

The Examiner's claim construction renders *storing first information with first data* as not being limited to the first information and the first data being stored in the same block. Particularly, the Examiner's position is that although he does not dispute that the term *with* and

[5] *Phillips v. AWH Corp.*, 75 USPQ2d 1321 (Fed. Cir. 2005)(en Banc); MPEP 2111.
[6] *Id.*
[7] *In re American Academy of Science Technical Center*, 70 USPQ2d 1827 (Fed. Cir. 2004); *In re Cortright*, 49 USPQ2d 1463, 1468 (Fed. Cir. 1999); *In re Morris*, 44 USPQ2d 1023 (Fed. Cir. 1997)
[8] *Phillips, supra*; *In re Morris, supra*; *In re Zletz*, 13 USPQ2d 1320, 1322 (Fed. Cir. 1989)

"together" might be somehow analogous,[9] he nonetheless believes that the two can be stored "with" each other yet reside in different blocks.

Thus, his rationale for the anticipatory rejection is that Lubbers '643 discloses user data "stored with" FE bits because both are depicted in the RAID array 10 of FIG. 2,[10] although he admits the data and FE bits are stored in different blocks.[11] Another rationale given is that Lubbers '643 discloses user data "stored with" FE bits because both are stored in the same disc drive.[12]

However, the Examiner has not provided any evidence whatsoever to rebut Applicant's arguments that the Examiner's interpretation is entirely inconsistent with the usage of the disputed phrase in the specification.[13] Reiterating, for example, the specification discloses the storing of metadata (*first information*) and user data (*first data*) only in terms of them being stored in the same block. FIG. 4A, for example, clearly sets forth the data block 400 having both a user data constituent 410 and a metadata constituent 420.[14] Furthermore, storing the metadata and user data in different blocks defeats the disclosed purpose of reducing the system performance overhead by retrieving both with a single access command instead of multiple commands.[15] Additionally, storing the metadata and user data in different blocks defeats the disclosed purpose of reducing the system processing overhead by eliminating the maintenance of mapping tables.[16]

---

[9] Examiner's Answer pg. 10 lines 13-14.
[10] Advisory Action of 10/24/2006; Office Action of 8/2/2006 pg. 8 lines 13-14.
[11] Examiner's Answer pg. 10 lines 20-22 ("Lubbers discloses a disk drive storing blocks of data...along with FE blocks that identify whether data blocks are considered bad....", emphasis added)
[12] Examiner's Answer pg. 11 lines 2-4.
[13] See Appellant's Brief ppg. 6-7; Applicant's Pre-Brief Request ppg. 1-2; Applicant's Response filed 10/2/2006 ppg. 7-9; Applicant's Response filed 7/5/2006 pg. 8
[14] See for example paragraph [0019] of the published application.
[15] See for example paragraphs [0021 and 0030].
[16] *Id.*

The Examiner has also provided no evidence whatsoever substantiating a reason that the skilled artisan, having read the specification, would reasonably conclude that *storing first information with first data* means storing them in different blocks. The reasons against the skilled artisan reaching such a conclusion are many.

First and fundamentally, as discussed above, the specification clearly requires that the user data and the metadata reside in the same block in order to achieve the disclosed purposes. Second and just as fundamentally, the skilled artisan has a working knowledge that the meaning of "storing data" includes a precise knowledge of an address within the storage space that the data is stored. The address is synonymous with fixed size chunks of storage capacity commonly referred to as data blocks. Particularly, the drive controller addresses data blocks in terms their unique cylinder-head-sector combination. Generally, however, the device driver and operating system, by which external devices store data, address the blocks via a layer of abstraction referred to as logical block addresses (LBAs). The skilled artisan understands before even reading the specification that a host access command to store data in a disc drive, such as SCSI or ATA drive, includes not just the user data to be stored but also the LBA where it is stored. Likewise, a host access command to subsequently retrieve stored data includes the LBA where the data is located.

Generally, the term "storing" plainly means putting something aside for subsequent retrieval. However, to the skilled artisan the term "storing" with respect to "storing data" implicitly requires knowledge of the LBA associated with the data. There is no reasonable rationale in the record for divorcing the concept of "storing data" from the LBA to which it is stored. Thus, *storing first information with first data* to the skilled artisan commonly means that both are stored to the same LBA.

The Examiner's unsubstantiated allegation that *storing first information with first data* could reasonably mean that they are stored in different LBAs appears to be the result of a misplaced attempt to conclude the skilled artisan would view the term "storing" of the present context in the same manner as in storing a household commodity like peanut butter. That is, "storing peanut butter" requires only a sealed container; unlike "storing data," the meaning of "storing peanut butter" does not require a precise definition of the place it is stored. For example, the meaning of "storing peanut butter" could plainly mean sealing the peanut butter in a jar, or it could equally mean putting the sealed jar in a cabinet. In other words, the peanut butter could reasonably be said to be "stored" either in the jar or the cabinet. Likewise, the meaning of "storing peanut butter and jelly" could plainly mean sealing the peanut butter and the jelly in separate jars, or it could equivalently mean sealing them together in a common jar,[17] or it could equivalently mean putting the jar or jars in a cabinet. Again, the peanut butter and jelly could reasonably be said to be "stored" either in the jar(s) or in the cabinet.

The Examiner attempts an analogous argument by postulating that the skilled artisan would view *storing first information with first data* means the two are in stored in different blocks but in the same disc drive (different jars but same cabinet). But the meaning of them both being stored in the same disc drive cannot be divorced from what LBA or LBAs they are stored in. The skilled artisan readily understands the concept of storing data is fundamentally different than storing peanut butter, such that the meaning of *with* in the context of *storing first information with first data* plainly means that both are stored together at the same LBA. The Examiner has simply acted as a sole arbiter and not substantiated any evidence, only his opinion, that the disputed phrase could reasonably mean that both are stored apart at different LBAs.

[17] Reference Goober Grape, http://www.smuckers.com/fg/pds/default.asp?groupid=2&catid=4&prodid=106

The Examiner's interpretation that *first information stored with first data* means they are stored in different blocks is inconsistent with both what the skilled artisan knows fundamentally about storing data and how he would view the usage of the disputed phrase in the specification, and thereby not within the broadest reasonable meaning consistent with the specification. Therefore, the Examiner has failed to substantiate anticipation because the cited reference does not disclose all the recited features of claim 1.

Accordingly, the evidence in the record supports Applicant's belief that the final rejection of claim 1 and the claims depending therefrom should be reversed.

Claims 8 and 15

These claims recite in pertinent parts:

accompanying first information with first data....[18]

first information accompanying first data....[19]

Applicant's stated position is that the term "accompanying" in the context of these claims means "joining" and is equivalent to the meaning that the *first information* and the *first data* reside together in the same block for the reasons set forth above for claim 1.[20]

For the reasons set forth above for claim 1, the Examiner's interpretation that these disputed phrases mean the *first information* and the *first data* are stored in different blocks is inconsistent with both what the skilled artisan knows fundamentally about storing data and how he would view the usage of the disputed phrase in the specification, and thereby not within the broadest reasonable meaning consistent with the specification. Therefore, the Examiner has

[18] Excerpt of claim 8, emphasis added.
[19] Excerpt of claim 15, emphasis added.
[20] See Appellant's Brief ppg. 9-11; Applicant's Pre-Brief Request ppg. 1-2; Applicant's Response filed 10/2/2006 ppg. 10-12; Applicant's Response filed 7/5/2006 ppg. 9-10.

failed to substantiate anticipation because the cited reference does not disclose all the recited features of claims 8 and 15.

Accordingly, the evidence supports Applicant's belief that the final rejection of claims 8 and 15 and the claims depending therefrom should be reversed.

THE SECTION 102 REJECTION OF INDEPENDENT CLAIMS 1, 8, AND 15 OVER MORGAN '411 IS CLEAR ERROR BECAUSE THE EVIDENCE IN THE RECORD IS DETERMINITIVE THAT THE REFERENCE DOES NOT DISCLOSE ALL THE RECITED FEATURES OF THESE CLAIMS

Claim 1

Claim 1 recites *storing first information with the first data, wherein the first information directly indicates the status of the first data.* Applicant's position is that the broadest reasonable interpretation of this disputed claim phrase is that the *first information* renders qualitative information about the *first data* to the extent that the *first data* is reliable or unreliable.

During examination claims are given their "broadest reasonable interpretation consistent with the specification."[21] The "broadest reasonable interpretation" is the meaning that the skilled artisan would give to the claim term in view of the associated usage provided in the specification.[22] A construction that is inconsistent with the written description would not be arrived at by the skilled artisan, and is therefore not a "reasonable interpretation."[23]

Preliminarily, the Examiner's claim construction is erroneous because it misstates the claimed subject matter:

> Since the code bits (first information) identifies the reliability or status of the configured data (first data), it directly indicates status of first data associated with the first data.[24]

[21] *Phillips v. AWH Corp., supra*; MPEP 2111
[22] *In re American Academy of Science Technical Center, supra; In re Cortright, supra; In re Morris, supra.*
[23] *Phillips, supra; In re Morris, supra; In re Zletz*, 13 USPQ2d 1320, 1322 (Fed. Cir. 1989)
[24] Examiner's Answer pg. 14 lines 9-11, emphasis added.

Giving the Examiner the benefit of the doubt for a typographical error, his claim construction effectively renders the *first information directly indicates the status of the first data* as being anticipated by the code bits in Morgan '411. The Examiner does not disagree that the code bits in Morgan '411 are used to determine which operation was occurring when a process failure occurs. For example:

> It should be understood that the setting and unsetting (re-setting) of the predetermined code bits refers to, respectively, setting the bits to preselected values, one per each operation whose failure is to be detected, and, unsetting the bits to some preselected value indicating no failed operation...Thus, determination can be made as to which data operation was taking place when the fault occurred.[25]

The Examiner also does not disagree that the code bits in Morgan '411 do not even directly indicate which operation was occurring when a failure occurred, but that the code bits are used to generate failure data via post-processing analyses. For example:

> The set code bits are accessed using error recovery routines and the accessed code bits are used in generating failure data that is utilized by service personnel in taking corrective action at step 106.[26]

The Examiner also does not disagree that the code bits in Morgan '411 are not used to preempt using unreliable data during a current operation and the concomitant cascading errors, but rather they simply mark the operation as a candidate for post-processing error analyses. For example:

> Data blocks are labelled with the identity of the fault operation while communication continues between the host system and the array storage system and any such fault indication can be later analyzed at a convenient time.[27]

[25] Morgan '411 col. 6 lines 31-52, emphasis added.
[26] Morgan '411 col. 6 lines 28-31, emphasis added.
[27] Morgan '411 col. 2 lines 63-67, emphasis added.

The Examiner's rationale for the anticipatory rejection is that "Morgan further discloses the code bits are reset to indicate the data transfer was successful, which indicates no fault in the configured data."[28] However, even within the context of the passage upon which the Examiner relies it is clear that the code bits play absolutely no role in directly indicating the status of the data. For example:

> If such a data transfer is occurring, predetermined code bits of the code byte are set at step 94 providing an indication that this data transfer is occurring. In step 98, a determination is made as to whether or not this data transfer was successfully completed. If successful, at step 102, the predetermined code bits that were previously set are now reset so that no inaccurate fault indication exists in the predetermined code bits. On the other hand, if the data transfer was not successfully completed, the code bits remain set.[29]

In Morgan '411 the code bits are set to indicate a data transfer operation is occurring. Some other signal, not disclosed but certainly not the code bits, directly indicates the status of the data to the code bits, *vis-à-vis* reporting a completed operation. If the other mechanism indicates that the transfer operation is unsuccessful, the code bits remain set to indirectly indicate a fault condition occurred during the transfer operation. If the other mechanism indicates that the transfer operation is successfully complete, the code bits are reset to indirectly indicate the transfer operation is no longer occurring. Incidentally, as the Examiner points out, if the other mechanism indicates that the transfer operation is successfully complete the resetting of the code bits can also be viewed as an indirect indication that no fault occurred in the transferred data. In either event the indication given by the code bits is indirect because it merely reflects the direct indication of the data provided by the other signal.

---

[28] Examiner's Answer pg. 7 lines 8-9, emphasis added.

[29] Morgan '411 col. 6 lines 18-28, emphasis added.

The Examiner has not substantiated any evidence that Morgan '411 discloses the code bits directly indicating the status of the data when unreliable data exists.[30] In that event the code bits only indicate what operation was occurring when a fault occurred, and the code bits must be recovered via ancillary processing steps, such as error correction routines, to generate other information indicating the status of the data.[31] Thus, the Examiner's interpretation of *first information directly indicates the status of first data* is that the code bits (*first information*) *directly* or indirectly *indicates* no unreliable data (*status of first data*) exists.

However, the Examiner has not provided any evidence whatsoever to rebut Applicant's arguments that the Examiner's interpretation is entirely inconsistent with the usage of the disputed phrase in the specification.[32] Reiterating, for example, the specification discloses the data reliability qualifier (DRQ) bits of the present embodiments are used to directly disclose the status of the user and parity data. The Examiner's interpretation defeats the disclosed purpose of discerning unreliable data in order to preempt its use in processing system commands. For example:

> As background, data are stored on a storage device such as a disc drive. The data may become corrupted because of physical defects on the media of the storage device. The data may also be corrupted for other reasons beside physically defective media. One example is when the data has been lost from a "write back" cache and which data was lost is known. Another example is when data cannot be reconstructed for an inoperative disk drive because the redundant copy is on physically defective media. In these cases, the data are not good or reliable even though the media where the data resides is not defective. Therefore, Data Reliability Qualifier (DRQ) bits are used to signal storage system 100 that the data are not reliable.

[30] Or even indirectly for that matter.
[31] Morgan '411 col. 6 lines 26-31.
[32] See Appellant's Brief ppg. 6-7; Applicant's Pre-Brief Request ppg. 1-2; Applicant's Response filed 10/2/2006 ppg. 7-9; Applicant's Response filed 7/5/2006 pg. 8

The storage system then can force an error message to the operating environment when the data is accessed.[33]

Another use of the present invention will be explained with reference to FIGS. 6A and 6B. FIG. 6A exemplifies when a device 620 is inoperative or "missing" in a disk array group 600. If a read request is made that resolves to device 620, the storage system controller receives data blocks P4, 11 and 12 from respective devices 610, 630 and 640. The controller will perform error detection of each block to ensure that the data is "good" (reliable from the point of view of the drive). If the data is "good", the storage system controller will exclusive-OR the parity data P4 in device 510 with data blocks 11 and 12 in respective devices 630 and 640.[34]

FIG. 6B shows when device 620 is inoperative and data block 12 of device 640 is unreliable. As described above, if a read request is made that accesses inoperative device 620, the storage system controller receives data blocks P4, 11 and 12 from respective devices 610, 630 and 640. The controller will perform error detection of each block to ensure that the data is "good". If any of the data is not "good", then the controller informs the host environment that the read cannot be performed.[35]

With further reference to FIG. 6B, writing data will be explained. In the case where data is to be written to block 10 of missing device 620 and block 12 of device 640 is unreadable (not "good"), the data in block 10 cannot be "stored in the parity data" in block P4 of device 610 because block 12 is unreadable. That is, the data in block 10 would normally be "stored" by generating a new parity block that is the exclusive-OR of the data in block 10 that is being written and the data in blocks 11 and 12. Normally, this situation results in a block that cannot be written. The data in block 12, however, can be made "good" by writing it with either "best guess" data or some pattern. The DRQ bit in the associated information data for block 12 will be set to "1" to remember that the data in block 12 is "unreliable". Now the data in block 10 can be "stored in the parity" because the data in block 12 has been "made good." The parity DRQ bit associated with parity block P4 will be

[33] Specification paragraph [0016], emphasis added.
[34] Specification paragraph [0025], emphasis added.
[35] Specification, paragraph [0026], emphasis added.

> generated using exclusive-OR from the new DRQ bit for data block 10, the existing DRQ bit for data block 11 and the set DRQ bit that represents the data in block 12 as unreliable.[36]
>
> FIG. 7 shows another use of the present invention. Shown is an array 700 that includes devices 710, 720, 730 and 740 configured as RAID 0. In other words, the data is striped but there is no parity. As such, the data is not recoverable. In the case where data block 14 (shown as the striped-out data block in device 720) is unreadable (not "good"), the data block is made readable again by either writing a "best guess" of the data in data block 14, or a pattern. Such a pattern can be all zeros. However, the data in data block 14 cannot be trusted and is, therefore, "unreliable." So the associated DRQ bit is set to indicate that the data in data block 14 is not trustworthy. The "Data Reliability Qualifier" should be understood as "logical metadata" associated with "logical blocks" even for RAID-0, where there is no redundancy.[37]

The Examiner has also provided no evidence whatsoever substantiating a reason that the skilled artisan, having read the specification, would reasonably conclude that *first information directly indicates the status of first data* could reasonably mean using the first information merely to indicate the status of the first data is reliable, as the Examiner suggests.

First and fundamentally, as discussed above, the specification clearly requires that the first information be capable of discerning the status of unreliable data in order to achieve the disclosed purposes. Second and just as fundamentally, the skilled artisan has a working knowledge that data reliability schemes generally, of which the present embodiments are a particular solution, are only valuable to the extent that they preempt the use of or at least indicate the presence of unreliable data. The Examiner's interpretation renders the *first information* completely superfluous.

[36] Specification paragraph [0027], emphasis added.

[37] Specification paragraph [0028], emphasis added.

The Examiner's interpretation that *first information directly indicates the status of the first data* means that it "indirectly indicates valid data" is inconsistent with the express language of the claim, inconsistent with what the skilled artisan knows fundamentally about storing data, inconsistent with how the skilled artisan would view the usage of the disputed phrase in the specification, and thereby not within the broadest reasonable meaning consistent with the specification. Therefore, the Examiner has failed to substantiate anticipation because the cited reference does not disclose all the recited features of claim 1.

Accordingly, the evidence in the record supports Applicant's belief that the final rejection of claim 1 and the claims depending therefrom should be reversed.

Claims 8 and 15

These claims recite in pertinent parts *accompanying first information with first data, wherein the first information indicates the status of second data associated with the first data.* Applicant's position is that the broadest reasonable interpretation of this disputed claim phrase means that the first information indicates the status of second data other than the first data that it accompanies.

As understood, the Examiner's claim construction renders *accompanying the first information with first data* as anticipated by block 74 in FIG. 2 of Morgan '411, which states "Appending code bits to each data block of converted data," and *second data* as anticipated by block 70 of FIG. 2 which states "Converting data from host system."

Appellant reiterates that the Examiner's construction is reversible error because it mischaracterizes one block of data in Morgan '411 as anticipating both the *first data* and the

second data of the recited claim language.[38] This is clear from the fact that the appending step 74 appends code bits to each block of converted data that was immediately converted in block 70. This is also clear from the fact that even the passage the Examiner relies upon explicitly states that the converting in block 70 and the appending in block 74 are done in conjunction for each block of data:

> In conjunction with converting the data to a suitable configuration, at step 74, a code byte having a number of code bits is appended to each block of the configured data.[39]

Appellant reiterates that the skilled artisan clearly understands that Morgan '411 discloses a one-to-one relationship between code bits and their respective blocks of user data. Contrary to the Examiner's claim construction, Morgan '411 discloses the same data block being both converted and appended. The Examiner's construction that converting a data block anticipates *first data* and appending the code bits anticipates *second data* is based on a mischaracterization of what Morgan '411 actually discloses.

As understood, the Examiner then construes the resetting of the code bits after a successful data transfer operation as anticipating *first information indicating the status of second data associated with the first data.* However, the skilled artisan clearly understands that successfully transferring a given block of data does not create two blocks of data; rather, it transfers the given block to a different block address.

Therefore, the Examiner has not substantiated any evidence that either the converting in conjunction with appending a block or the transferring a block anticipates the *second data associated with the first data* as claimed. At best, the changed code bits after a successful data

[38] See Appellant's Brief pg. 16; Applicant's Pre-Brief Request pg. 4; Applicant's Response filed 10/2/2006 pg. 16.
[39] Morgan '411 col. 5 lines 56-59, emphasis added.

transfer in Morgan '411 might be viewed as "second information indirectly indicating the status of first data," but such pontification is irrelevant in view of the claimed subject matter.

The Examiner's claim construction is reversible error because it depends upon a mischaracterization of what Morgan '411 actually discloses. Therefore, the Examiner has failed to substantiate anticipation because the cited reference does not disclose all the recited features of claims 8 and 15.

Accordingly, the evidence in the record supports Applicant's belief that the final rejection of claims 8 and 15 and the claims depending therefrom should be reversed.

Conclusion

In conclusion, Applicant respectfully submits that the Examiner has not substantiated an anticipatory rejection for any of the independent claims. Accordingly, the Applicant respectfully requests that the final rejection of the independent claims and the claims depending therefrom be reversed.

Respectfully submitted,

By: _____

Mitchell K. McCarthy, Reg. No. 38,794
Randall K. McCarthy, Reg. No. 39,297
Fellers, Snider, Blankenship, Bailey & Tippens
100 North Broadway, Suite 1700
Oklahoma City, Oklahoma 73102
Telephone: (405) 232-0621
Facsimile: (405) 232-9659
Customer No. 33900